# Team T

- Hemanth Thathireddy (A20525346)
- Satya Dineswara Reddy Satti (A20528639)
- Sai Manasa Basani (A20544372)
- Alavalapati Veera manohar Reddy (A20526070)
- Ying Zhang (A20547843)

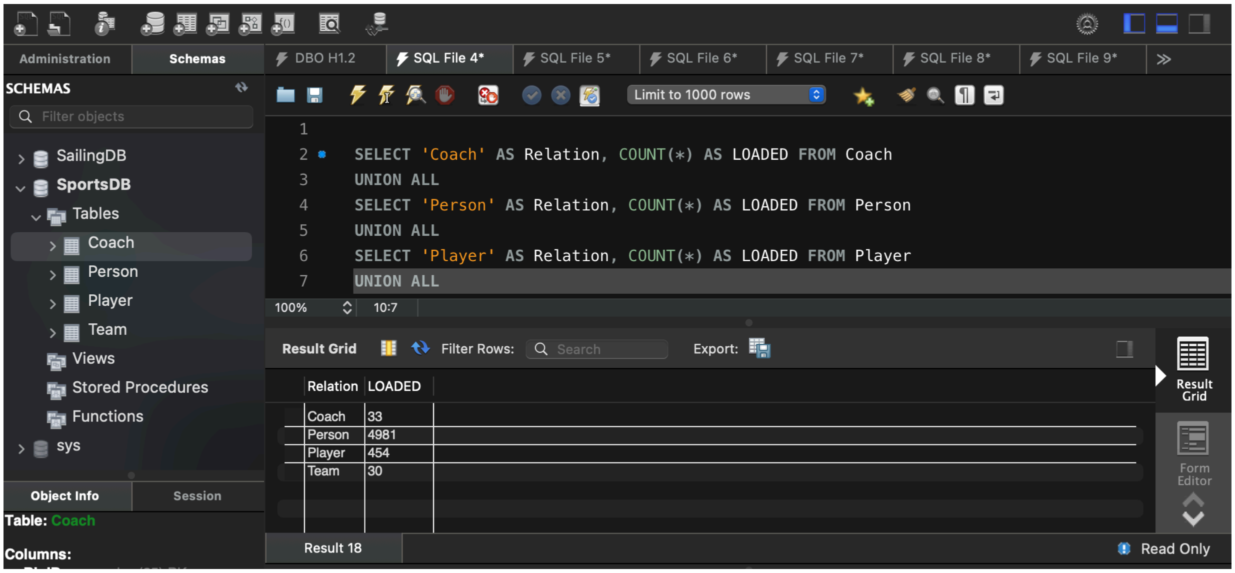
- **The query statement**

1. Output the number of rows in each of your 4 relations (using 4 SELECT statements) in this order: Coach, Person, Player, Team. Call the result column LOADED each time.

- **The SQL commands** :

```
SELECT 'Coach' AS Relation, COUNT(*) AS LOADED FROM Coach
UNION ALL
SELECT 'Person' AS Relation, COUNT(*) AS LOADED FROM Person
UNION ALL
SELECT 'Player' AS Relation, COUNT(*) AS LOADED FROM Player
UNION ALL
```

- **Query result/output**



```

SELECT 'Coach' AS Relation, COUNT(*) AS LOADED FROM Coach
UNION ALL
SELECT 'Person' AS Relation, COUNT(*) AS LOADED FROM Person
UNION ALL
SELECT 'Player' AS Relation, COUNT(*) AS LOADED FROM Player
UNION ALL
```

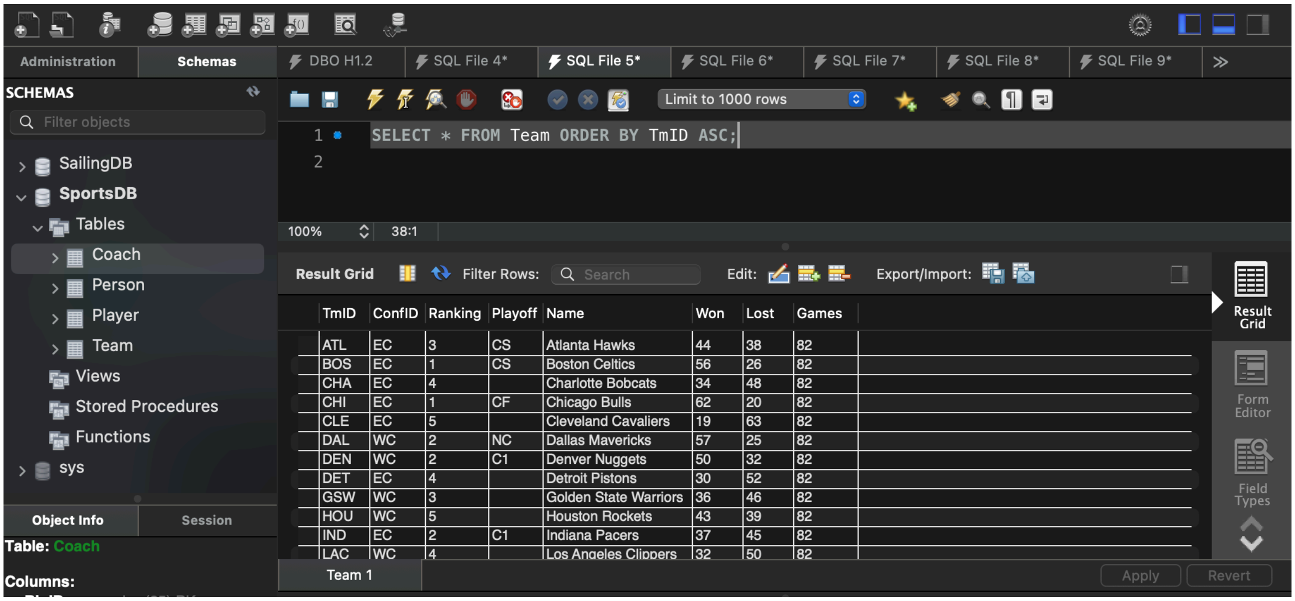
| Relation | LOADED |
|---|---|
| Coach | 33 |
| Person | 4981 |
| Player | 454 |
| Team | 30 |

Table: Coach

Columns:

- **The query statement**

2. Output everything in the Team relation, in ascending (increasing) order of TmID.
   - **The SQL commands** : SELECT * FROM Team ORDER BY TmID ASC;
   - **Query result/output**

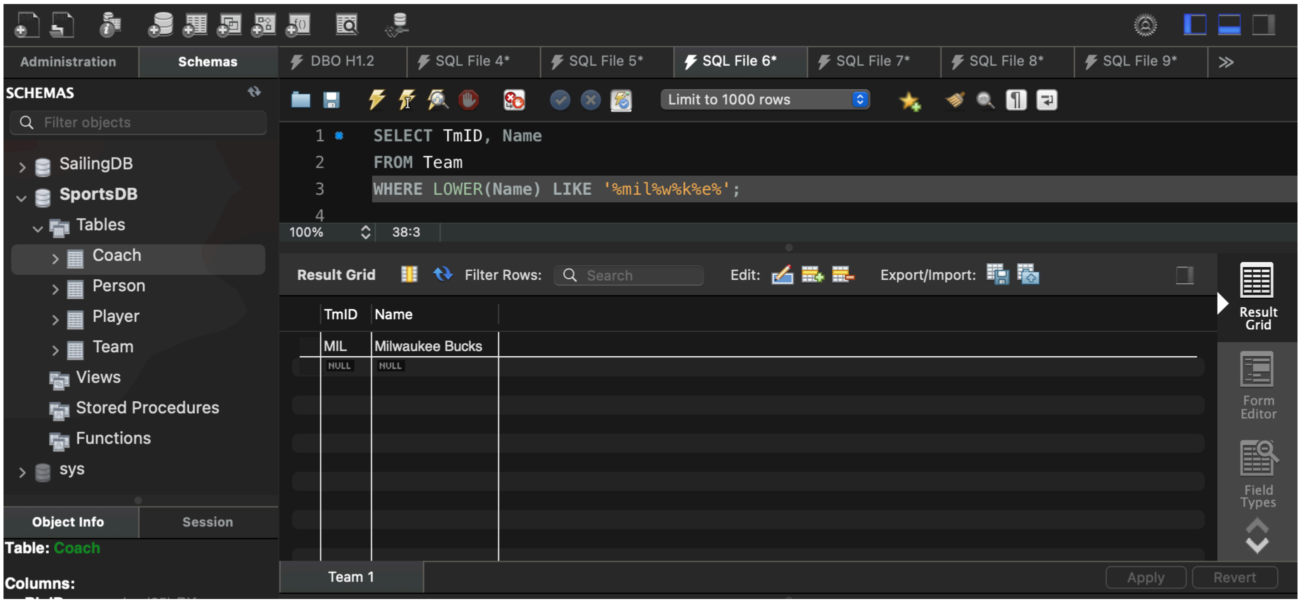


```
SELECT * FROM Team ORDER BY TmID ASC;
```

| TmID | ConfID | Ranking | Playoff | Name | Won | Lost | Games |
|---|---|---|---|---|---|---|---|
| ATL | EC | 3 | CS | Atlanta Hawks | 44 | 38 | 82 |
| BOS | EC | 1 | CS | Boston Celtics | 56 | 26 | 82 |
| CHA | EC | 4 |  | Charlotte Bobcats | 34 | 48 | 82 |
| CHI | EC | 1 | CF | Chicago Bulls | 62 | 20 | 82 |
| CLE | EC | 5 |  | Cleveland Cavaliers | 19 | 63 | 82 |
| DAL | WC | 2 | NC | Dallas Mavericks | 57 | 25 | 82 |
| DEN | WC | 2 | C1 | Denver Nuggets | 50 | 32 | 82 |
| DET | EC | 4 |  | Detroit Pistons | 30 | 52 | 82 |
| GSW | WC | 3 |  | Golden State Warriors | 36 | 46 | 82 |
| HOU | WC | 5 |  | Houston Rockets | 43 | 39 | 82 |
| IND | EC | 2 | C1 | Indiana Pacers | 37 | 45 | 82 |
| LAC | WC | 4 |  | Los Angeles Clippers | 32 | 50 | 82 |

- **The query statement**

3. Show the TmID of teams with Milwaukee in their name, but your query must cater for the fact that people spell this incorrectly – everyone starts with “MIL”, then somewhere later they have a “W” and even later a “K”. Example misspellings are Millwaukee, Milwakee, Milwuakee, Milwaukey, etc.
   - **The SQL commands** :
     SELECT TmID, Name FROM Team
     WHERE LOWER(Name) LIKE '%mil%w%k%e%';
   - **Query result/output**

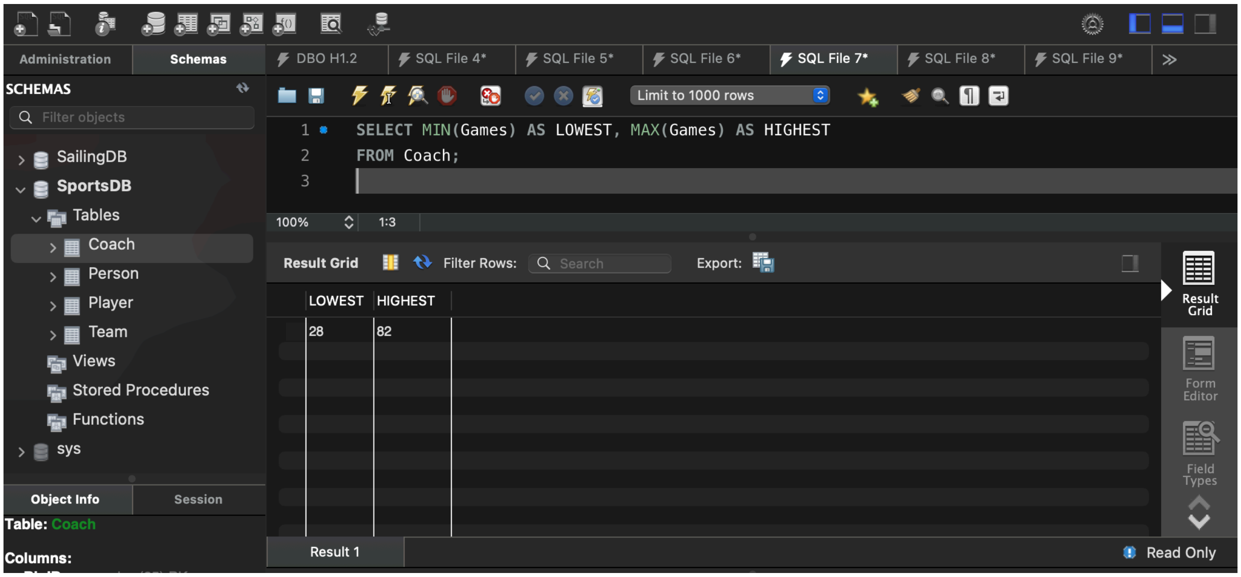


```
SELECT TmID, Name
FROM Team
WHERE LOWER(Name) LIKE '%mil%w%k%e%';
```

| TmID | Name |
|---|---|
| MIL | Milwaukee Bucks |
| NULL | NULL |

- **The query statement**

4. What are the lowest and the highest number of Games (played) in the coach data? Call the first result column LOWEST and the second result column HIGHEST.
   - **The SQL commands** : SELECT MIN(Games) AS LOWEST, MAX(Games) AS HIGHEST FROM Coach;
   - **Query result/output**

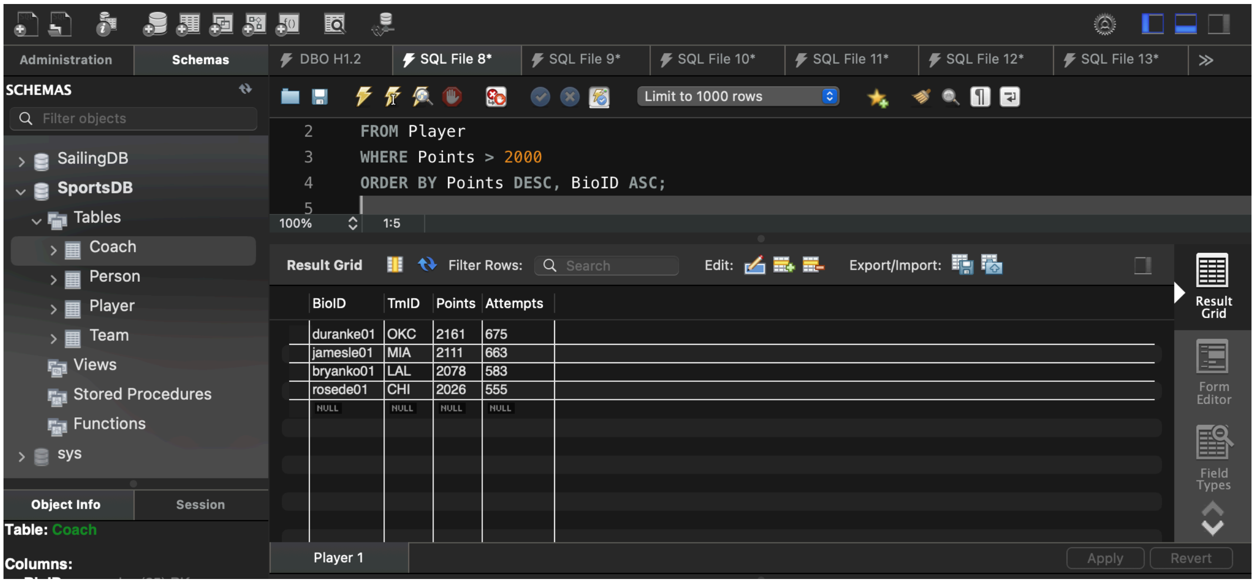


```
SELECT MIN(Games) AS LOWEST, MAX(Games) AS HIGHEST
FROM Coach;
```

| LOWEST | HIGHEST |
|---|---|
| 28 | 82 |

- **The query statement**

5. Show the BioID, TmID, Points and Attempts for each player that scored more than 2000 points, in decreasing order of Points. Players with the same number of Points should be shown in alphabetical order of BioID.
   - **The SQL commands** : SELECT BioID, TmID, Points, Attempts FROM Player WHERE Points > 2000 ORDER BY Points DESC, BioID ASC;
   - **Query result/output**



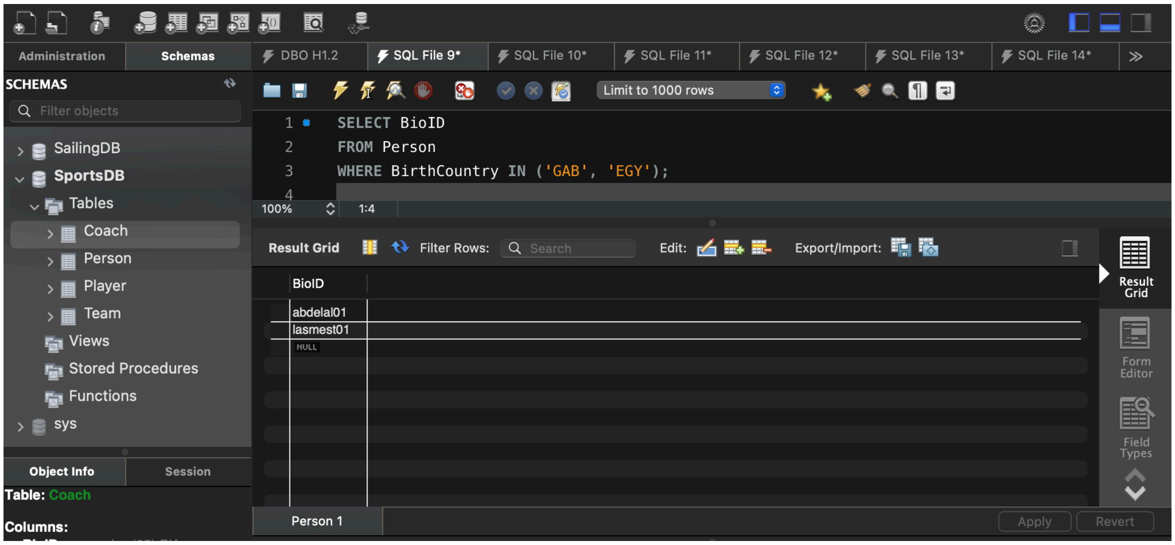
```
FROM Player
WHERE Points > 2000
ORDER BY Points DESC, BioID ASC;
```

| BioID | TmID | Points | Attempts |
|---|---|---|---|
| duranke01 | OKC | 2161 | 675 |
| jamesle01 | MIA | 2111 | 663 |
| bryanko01 | LAL | 2078 | 583 |
| rosede01 | CHI | 2026 | 555 |
| NULL | NULL | NULL | NULL |

- **The query statement**

6. Show BioID of persons born in Gabon (GAB) and BioID of persons born in Egypt (EGY), if any.

- **The SQL commands** : SELECT BioID FROM Person WHERE BirthCountry IN ('GAB', 'EGY');
- **Query result/output**



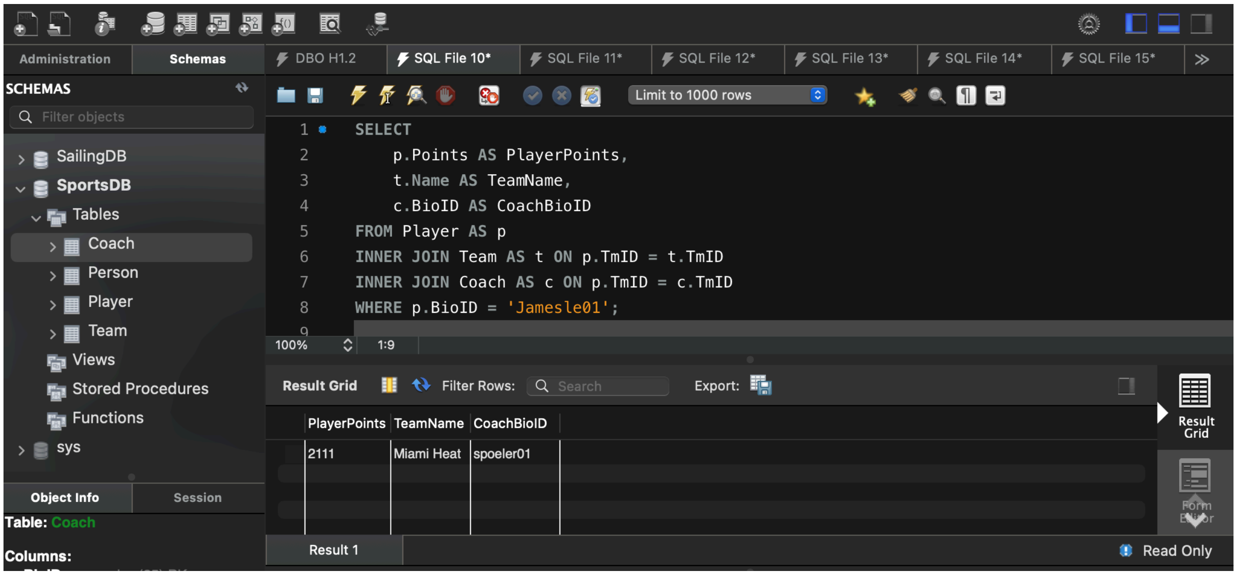
```
SELECT BioID
FROM Person
WHERE BirthCountry IN ('GAB', 'EGY');
```

| BioID |
|---|
| abdelal01 |
| lasmest01 |
| NULL |

- **The query statement**

7. Show the following for Jamesle01: his Points, the full Name of his team, & the BioID of his team's coach.

- **The SQL commands:** SELECT p.Points AS PlayerPoints, t.Name AS TeamName, c.BioID AS CoachBioID FROM Player AS p INNER JOIN Team AS t ON p.TmID = t.TmID INNER JOIN Coach AS c ON p.TmID = c.TmID WHERE p.BioID = 'Jamesle01';
- **Query result/output**



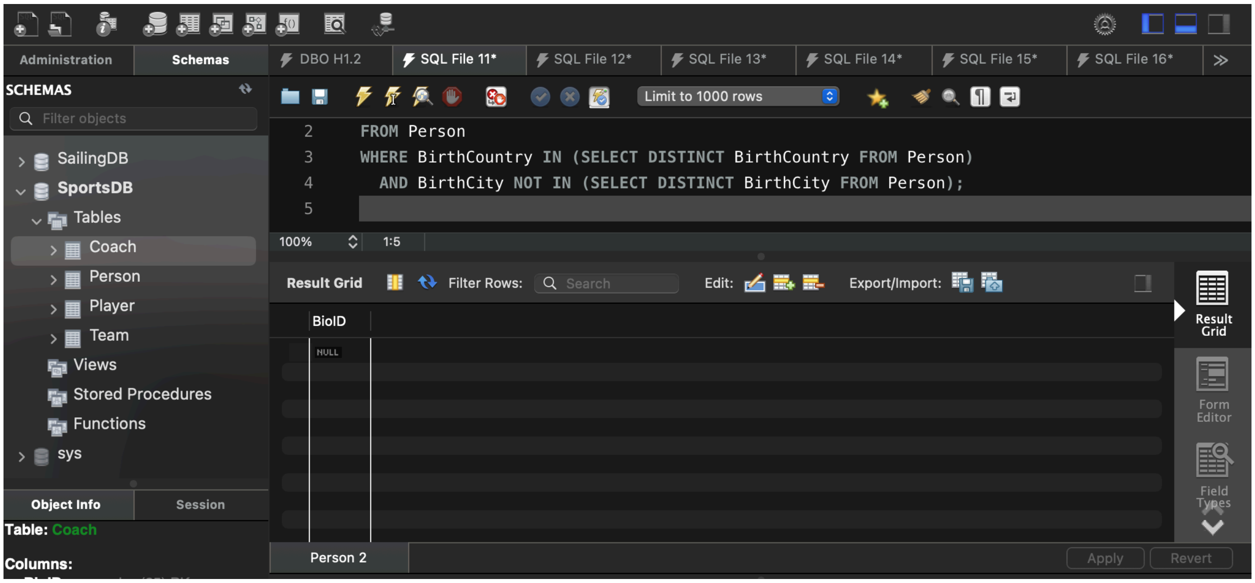
```
SELECT
    p.Points AS PlayerPoints,
    t.Name AS TeamName,
    c.BioID AS CoachBioID
FROM Player AS p
INNER JOIN Team AS t ON p.TmID = t.TmID
INNER JOIN Coach AS c ON p.TmID = c.TmID
WHERE p.BioID = 'Jamesle01';
```

| PlayerPoints | TeamName | CoachBioID |
|---|---|---|
| 2111 | Miami Heat | spoeler01 |

- **The query statement**

8. Show the BioID of players whose BirthCountry is in the data, but their BirthCity is not in the data.

- **The SQL commands** : FROM Person WHERE BirthCountry IN (SELECT DISTINCT BirthCountry FROM Person) AND BirthCity NOT IN (SELECT DISTINCT BirthCity FROM Person);
- **Query result/output**



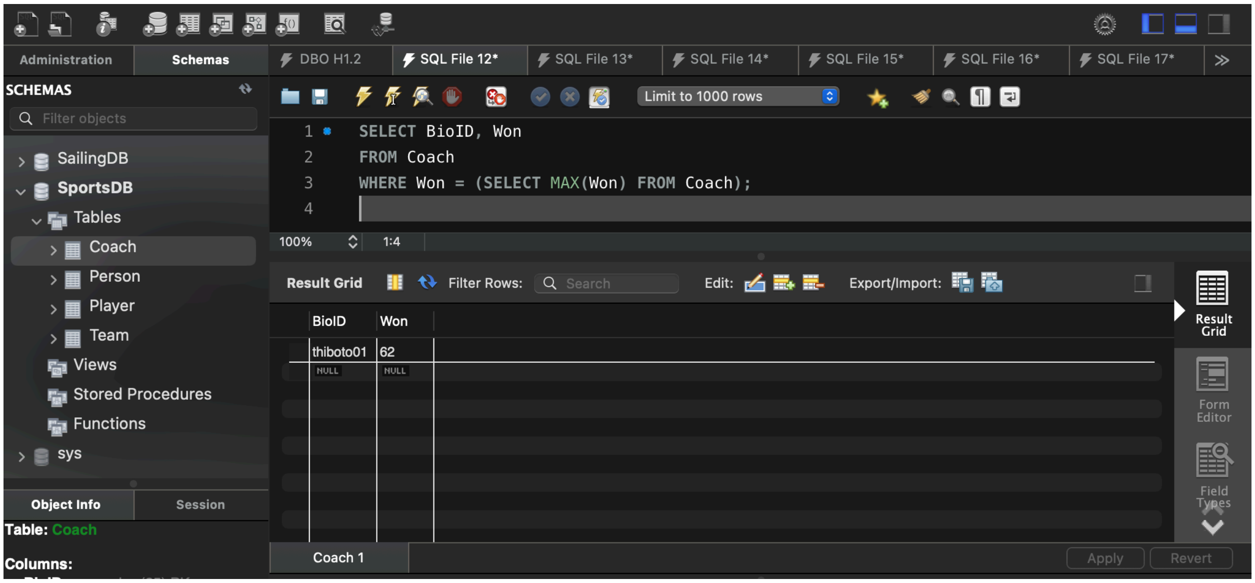
```
FROM Person
WHERE BirthCountry IN (SELECT DISTINCT BirthCountry FROM Person)
  AND BirthCity NOT IN (SELECT DISTINCT BirthCity FROM Person);
```

```
BioID
NULL
```

- **The query statement**

9. Which coach(es) Won the most games? Give BioID and the number Won.

- **The SQL commands:** SELECT BioID, Won FROM Coach WHERE Won = (SELECT MAX(Won) FROM Coach);
- **Query result/output**



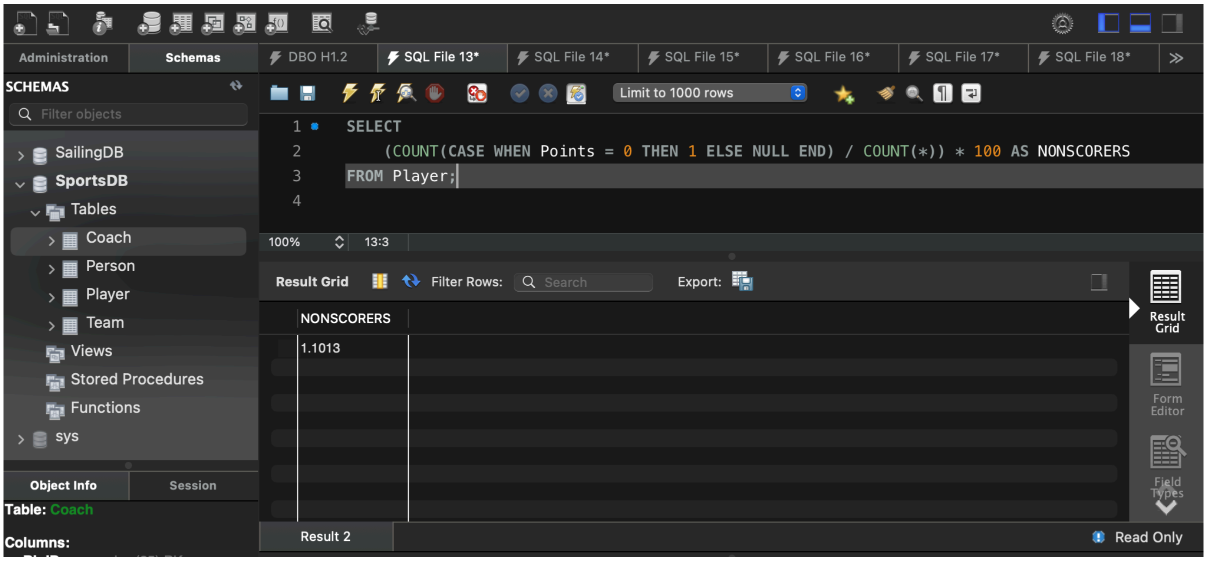
```
SELECT BioID, Won
FROM Coach
WHERE Won = (SELECT MAX(Won) FROM Coach);
```

```
BioID      Won
thiboto01  62
NULL       NULL
```

- **The query statement**

10. What percentage of players have Points scored as zero? Call the result column NONSCORERS.

- **The SQL commands:** SELECT (COUNT(CASE WHEN Points = 0 THEN 1 ELSE NULL END) / COUNT(*)) * 100 AS NONSCORERS FROM Player;
- **Query result/output**



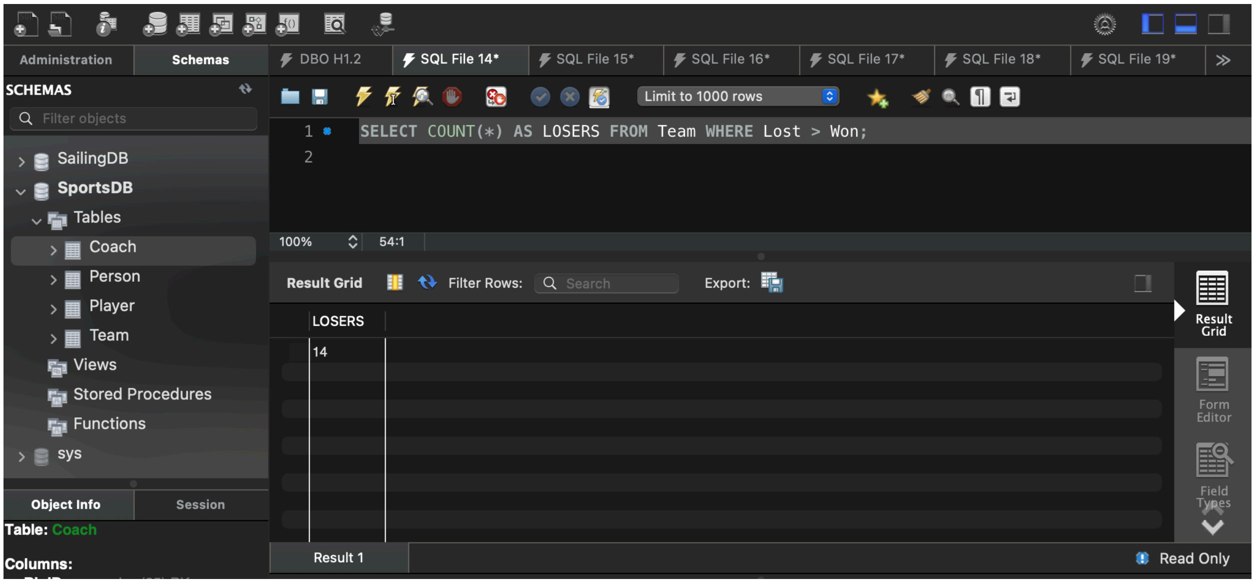
```
SELECT
    (COUNT(CASE WHEN Points = 0 THEN 1 ELSE NULL END) / COUNT(*)) * 100 AS NONSCORERS
FROM Player;
```

```
NONSCORERS
1.1013
```

- **The query statement**

11. How many teams have Lost more games than they have Won? Call the result column LOSERS.

- **The SQL commands:** SELECT COUNT(*) AS LOSERS FROM Team WHERE Lost > Won;
- **Query result/output**

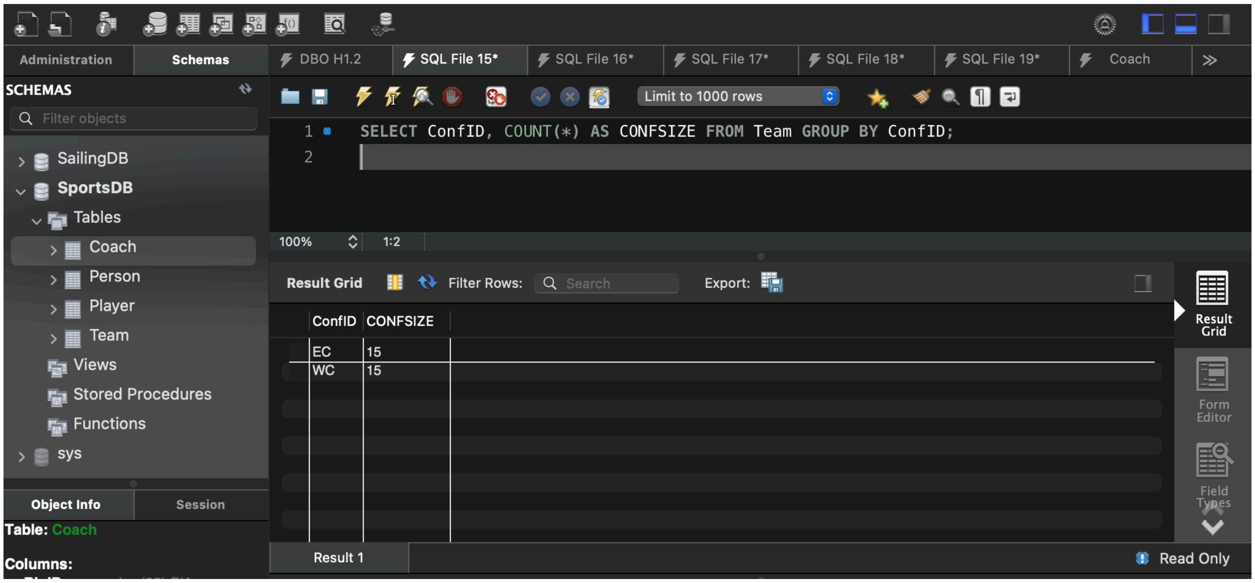


```
SELECT COUNT(*) AS LOSERS FROM Team WHERE Lost > Won;
```

```
LOSERS
14
```

- **The query statement**

12. How many teams belong to each ConfID? Call the 2nd column CONFSIZE.
    - **The SQL commands :** SELECT ConfID, COUNT(*) AS CONFSIZE FROM Team GROUP BY ConfID;
    - **Query result/output**

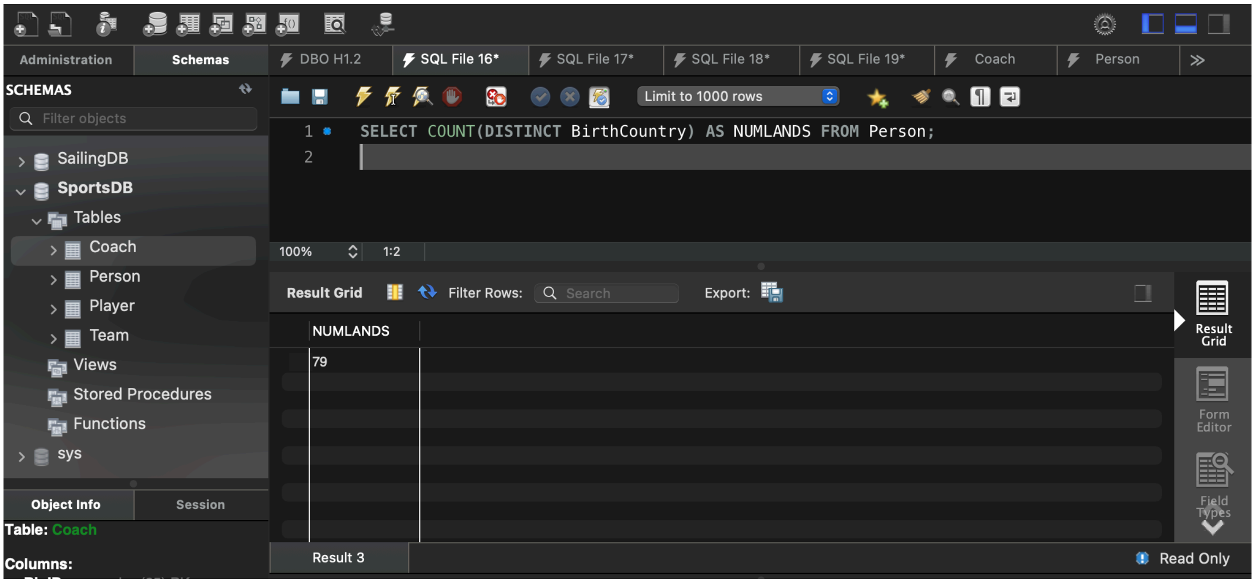


```
SELECT ConfID, COUNT(*) AS CONFSIZE FROM Team GROUP BY ConfID;
```

```
ConfID  CONFSIZE
EC      15
WC      15
```

- **The query statement**

13. How many countries do the persons in this data come from? Call the result column NUMLANDS.
    - **The SQL commands:** SELECT COUNT (DISTINCT BirthCountry) AS NUMLANDS FROM Person;
    - **Query result/output:**



```
SELECT COUNT(DISTINCT BirthCountry) AS NUMLANDS FROM Person;
```

```
NUMLANDS
79
```

14.

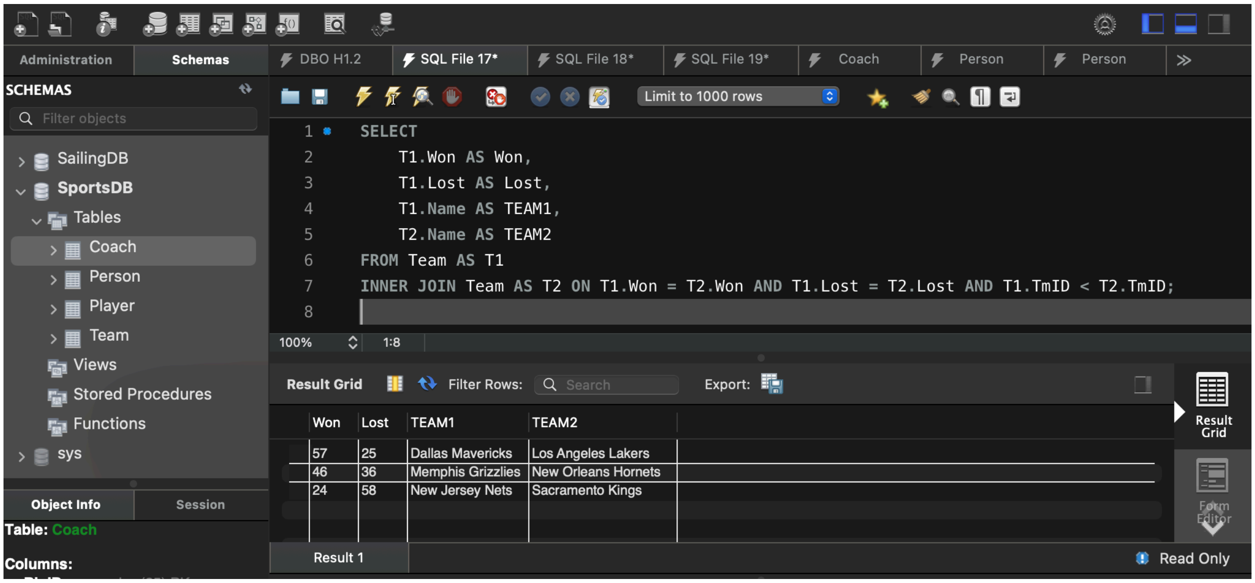
- **The query statement**

Which pairs of teams have the exact same record (meaning the same values for Won and the same values for Lost)? Show their Won value and then their Lost value and then the 2 Names, making sure that the 3rd column Name is alphabetically before the 4th column Name so information is not repeated. Call the 3rd column TEAM1 and the 4th column TEAM2.

- **The SQL commands :** SELECT DISTINCT

```
    t1.Won AS Won,
    t1.Lost AS Lost,
     t1.Name AS TEAM1,
    t2.Name AS TEAM2
FROM Team t1
JOIN Team t2 ON t1.Won = t2.Won AND t1.Lost = t2.Lost
WHERE t1.Name < t2.Name;
```

- **Query result/output**

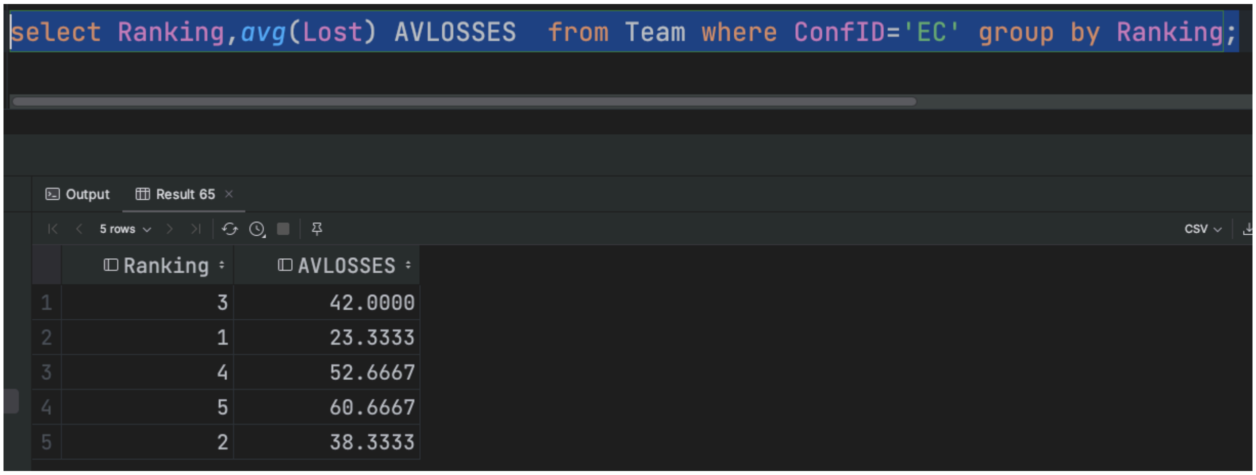


```
SELECT
    T1.Won AS Won,
    T1.Lost AS Lost,
    T1.Name AS TEAM1,
    T2.Name AS TEAM2
FROM Team AS T1
INNER JOIN Team AS T2 ON T1.Won = T2.Won AND T1.Lost = T2.Lost AND T1.TmID < T2.TmID;
```

| Won | Lost | TEAM1 | TEAM2 |
|---|---|---|---|
| 57 | 25 | Dallas Mavericks | Los Angeles Lakers |
| 46 | 36 | Memphis Grizzlies | New Orleans Hornets |
| 24 | 58 | New Jersey Nets | Sacramento Kings |

15. For each of the 5 Rankings in ConfID “EC”, show the average number of games Lost by those teams. Call the 2nd column AVLOSSES.

- **The SQL commands :**select Ranking,avg(Lost) AVLOSSES  from Team where ConfID='EC' group by Ranking;
- **Query result/output**



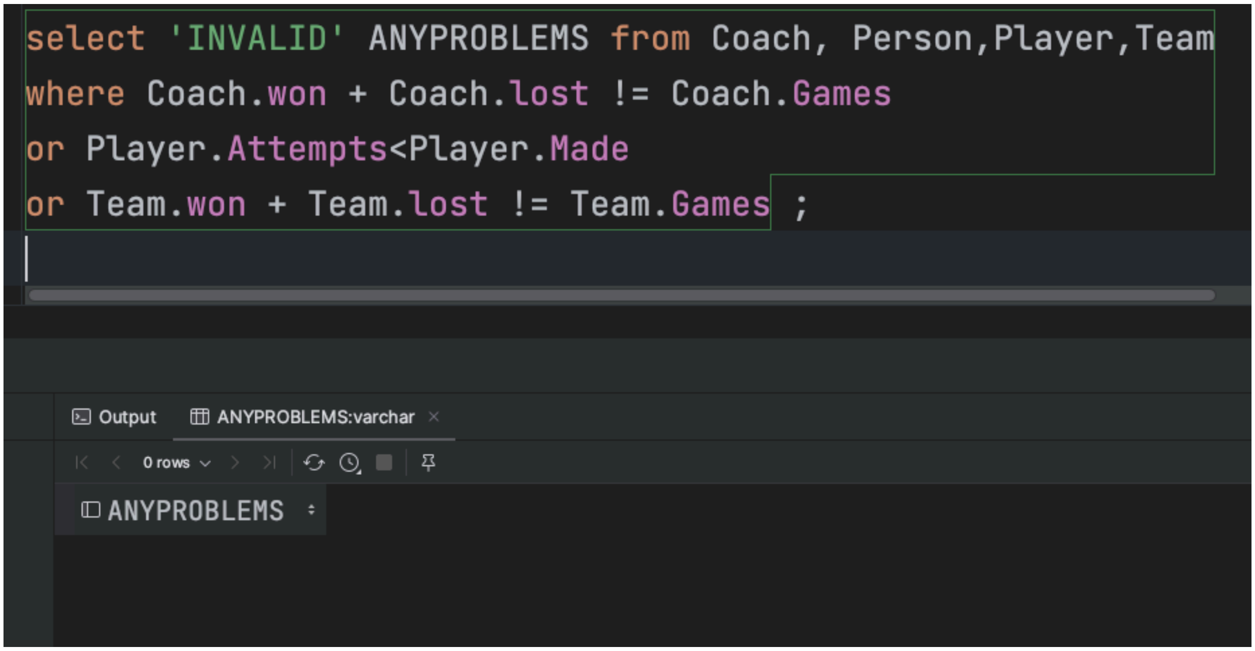
```
select Ranking,avg(Lost) AVLOSSES  from Team where ConfID='EC' group by Ranking;
```

| | Ranking | AVLOSSES |
|---|---|---|
| 1 | 3 | 42.0000 |
| 2 | 1 | 23.3333 |
| 3 | 4 | 52.6667 |
| 4 | 5 | 60.6667 |
| 5 | 2 | 38.3333 |

- **The query statement**

16. Give a SQL statement to output “INVALID” if any information in any row (tuple) of any relation has invalid data (e.g., the Games value is not equal to Won value plus Lost value). If all the data is valid then it should output an empty table. Call the result column ANYPROBLEMS.

- **The SQL commands :**select 'INVALID' ANYPROBLEMS from Coach, Person,Player,Team where Coach.won + Coach.lost != Coach.Games or Player.Attempts<Player.Made or Team.won + Team.lost != Team.Games ;
- **Query result/output**



```
select 'INVALID' ANYPROBLEMS from Coach, Person,Player,Team
where Coach.won + Coach.lost != Coach.Games
or Player.Attempts<Player.Made
or Team.won + Team.lost != Team.Games ;
```

| ANYPROBLEMS |
|---|

## Team Contributions:

| S.No | Name | Contribution(%) |
|---|---|---|
| 1. | Hemanth Thathireddy | 20% |
| 2. | Satya Dineswara Reddy Setti | 20% |
| 3. | Sai Manasa Basani | 20% |
| 4. | Alavalapati Veera manohar Reddy | 20% |
| 5. | Ying Zhang | 20% |